

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Departament de llenguatges i sistemes informàtics

**ACTUALITZACIÓ CONSISTENT DE
BASES DE DADES DEDUCTIVES**

Autor: Enric Mayol Sarroca
Director: Ernest Teniente i López

Barcelona, 2000

2. Base de Dades Deductiva

Una base de dades deductiva D està formada per tres conjunts finits: un conjunt F de fets, un conjunt R de regles deductives i un conjunt I de restriccions d'integritat. Un fet és un àtom de la base. El conjunt de fets s'anomena *base de dades extensional* (EDB) i el conjunt de regles deductives i restriccions d'integritat s'anomena *base de dades intensional* (IDB).

Suposarem que els predicats de la base de dades es divideixen en predicats bàsics i predicats derivats. Un predicat bàsic apareix a la base de dades extensional i, eventualment, en el cos de les regles deductives o de les restriccions d'integritat. Un predicat derivat apareix només a la base de dades intensional. Qualsevol base de dades deductiva es pot definir d'aquesta manera [BR86]. Els fets derivats no s'emmagatzemen explícitament a la base de dades extensional, sinó que es calculen a partir dels fets bàsics emmagatzemats a la EDB i utilitzant les regles deductives de la IDB.

A més a més, suposarem que cada predicat de la base de dades té associada una clau primària que representem amb els arguments \mathbf{k} . Així doncs, podem distingir dos tipus de predicats: els predicats $P(\mathbf{k}, \mathbf{x})$ amb arguments clau i no clau i, els predicats $P(\mathbf{k})$ amb només arguments clau. Ambdós \mathbf{k} i \mathbf{x} són vectors de variables. A la secció 2.2 formalitzarem el concepte de clau d'un predicat mitjançant una restricció d'integritat.

2.1 Regles deductives

Una regla deductiva o regla de derivació és una fórmula del tipus:

$$P \leftarrow L_1 \wedge \dots \wedge L_n \quad \text{amb } n \geq 1$$

on P és un àtom i els L_1, \dots, L_n són literals (és a dir, àtoms o àtoms negats). Totes les variables en P, L_1, \dots, L_n estan universalment quantificades davant la fórmula. 'P' rep el nom de *conclusió* i els literals L_1, \dots, L_n componen *el cos* de la regla. Suposarem que els termes de la conclusió són variables diferents i que els termes dels literals del cos poden ser variables o constants. Un predicat derivat pot estar definit per més d'una regla deductiva.

Els predicats del cos de la regla poden ser ordinaris o avaluable. Els primers són predicats bàsics o derivats, mentre que els darrers són predicats que en ser avaluats no requereixen accedir a la base de dades, com per exemple els predicats de comparació.

Com és habitual, requerim que la base de dades sigui *permissible* (en anglès, *allowed*) abans i després de qualsevol modificació [Llo87]. És a dir, qualsevol variable que ocorre en una regla deductiva ha de tenir una ocurrència en un literal positiu ordinari del cos de la regla. Aquesta condició assegura que tots els literal negats del cos es puguin instanciar completament abans de ser avaluats per la regla de negació per fracàs finit.

En aquesta tesi, considerem que la base de dades és *estratificada* [ABW88]. Una base de dades és estratificada si el conjunt dels seus símbols de predicat es pot dividir en un conjunt finit de classes, diguem S_0, \dots, S_n , de forma que per a cada regla deductiva $P \leftarrow L_1 \wedge \dots \wedge L_n$ amb $P \in S_j$:

- (i) si $Q \in S_i$ és el símbol de predicat d'un literal positiu del cos, llavors $i \leq j$, i
- (ii) si $Q \in S_i$ és el símbol de predicat d'un literal negatiu del cos, llavors $i < j$.

2.2 Restriccions d'integritat

Una restricció d'integritat és una fórmula tancada de primer ordre que qualsevol estat de la base de dades ha de satisfer. En el nostre enfocament, considerem que les restriccions d'integritat estan expressades en forma de *denegació*:

$$\leftarrow L_1 \wedge \dots \wedge L_n \quad \text{amb } n \geq 1 \quad (1)$$

on els L_i són literals i totes les variables se suposen universalment quantificades davant de la fórmula.

Restriccions d'integritat més generals poden expressar-se en forma de denegació. Per fer-ho, primer cal aplicar la transformació en *forma de rang* [Dec89] i, a continuació, utilitzar el procediment descrit a [LIT84]. La transformació en forma de rang és necessària perquè evita que la segona transformació introdueixi *entrebanc* (en anglès, *floundering*) en alguns casos. Una avaluació s'entrebanca si no es pot continuar el procés de derivació pel fet d'haver assolit un objectiu format únicament per literals negatius no instanciats.

Per motius d'uniformitat, associem a cada restricció d'integritat un predicat d'inconsistència Ic_n (de la mateixa manera que [Kow78]). Així, les restriccions tindran la mateixa forma que les regles deductives i les anomenarem *regles d'integritat*. Per exemple, la denegació anterior (1) es reescriurà com $Ic_1 \leftarrow L_1 \wedge \dots \wedge L_n$. De la mateixa manera que les regles de derivació, les regles d'integritat també han de ser permissibles.

Per tal de referir-nos a totes les restriccions d'integritat definides a la base de dades, suposarem que hi ha definit un predicat d'inconsistència Ic definit per les següents regles d'integritat:

$$\begin{aligned} Ic &\leftarrow Ic_1 \\ &\dots \\ Ic &\leftarrow Ic_n \end{aligned}$$

on n és el nombre de restriccions d'integritat definides a la base de dades.

En general, es poden diferenciar dos tipus de restriccions d'integritat: les *estàtiques*, que són aquelles restriccions d'integritat que s'han de satisfer en qualsevol estat de la base de dades i, les *de transició*, que són aquelles que s'han de satisfer durant la transició entre dos estats

consecutius de la base de dades. En aquesta tesi, ens centrarem en el primer tipus de restriccions, encara que el mètode proposat també pot tractar el segon tipus de restricció.

Per tal d'imposar el concepte de clau, suposarem que cada predicat bàsic o derivat $P(\mathbf{k}, \mathbf{x})$ té associada una restricció d'integritat de clau definida de la forma següent:

$$Ic_n(\mathbf{k}, \mathbf{x}, \mathbf{x}') \leftarrow P(\mathbf{k}, \mathbf{x}) \wedge P(\mathbf{k}, \mathbf{x}') \wedge \mathbf{x} \neq \mathbf{x}'$$

La clau d'un predicat derivat és deduïble a partir de les regles de derivació associades i de les claus dels predicats bàsics que el defineixen. El mecanisme per a deduir la clau d'un predicat derivat està basat en una adaptació [BFU92] del procediment descrit a [Dat90]. En aquesta tesi, suposarem que una actualització sempre satisfà les restriccions d'integritat associades a les claus així obtingudes.

Les restriccions de clau no es definiran com a restriccions explícites en la definició de la base de dades. En la definició dels predicats bàsics caldrà indicar quins arguments en componen la clau del predicat, en canvi, en el cas dels predicats derivats, les claus s'obtidran a partir del procediment definit a [BFU92]. Tenint en compte la informació de les claus, el mètode proposat en aquesta tesi, serà l'encarregat d'assegurar que les restriccions d'integritat de clau associades a tots els predicats de la base de dades es satisfan en tot moment.

Per qüestions de claredat, els arguments clau de tot predicat s'indicaran subratllats per a diferenciar-los dels arguments no clau.

Exemple 2.1: La base de dades deductiva següent proporciona informació sobre la situació laboral de les persones.

Predicats bàsics:

Prop (<u>p</u> , c)	La persona 'p' és propietària de l'empresa 'c'
Treb (<u>p</u> , c)	La persona 'p' treballa a l'empresa 'c'
Cont(<u>p</u> , c)	La persona 'p' té un contracte amb l'empresa 'c'
Baixa(<u>p</u>)	La persona 'p' està de baixa per malaltia
Edat(<u>p</u>)	La persona 'p' està en edat laboral
Sou(<u>p</u> , c, s)	La persona 'p' cobra un sou de 's' euros a l'empresa 'c'
Numss(<u>p</u> , n)	La persona 'p' té el número 'n' d'afiliació a la SS

Regles deductives:

$N\grave{o}mina(\underline{p}, c) \leftarrow Prop(\underline{p}, c)$
$N\grave{o}mina(\underline{p}, c) \leftarrow Treb(\underline{p}, c) \wedge \neg Baixa(\underline{p})$
$Emp(\underline{p}, c) \leftarrow Treb(\underline{p}, c) \wedge Cont(\underline{p}, c)$
$Actiu(\underline{p}) \leftarrow Emp(\underline{p}, c) \wedge \neg Baixa(\underline{p})$
$Contractat(\underline{p}) \leftarrow Cont(\underline{p}, c)$

Restriccions d'integritat:

$$Ic_1(\underline{p}, c) \leftarrow Emp(\underline{p}, c) \wedge \neg Edat(\underline{p})$$

$$Ic_2(p, c, s) \leftarrow Sou(p, c, s) \wedge \neg Treb(p, c)$$

$$Ic_3(p, n) \leftarrow Numss(p, n) \wedge \neg Contractat(p)$$

Observi's que en aquest exemple, el predicat derivat *Nòmina* està definit per més d'una regla de derivació. Aquest predicat estableix que una persona 'p' està en nòmina d'una empresa 'c' si és ell mateix el propietari de l'empresa, o si és una persona que hi treballa i no està de baixa. El predicat *Emp* estableix que una persona és un empleat d'una empresa si hi treballa i si té un contracte de treball amb aquesta. El predicat derivat *Actiu* defineix que tota persona empleada en una empresa i que no està de baixa per malaltia, és una persona activa. Finalment, el predicat derivat *Contractat* estableix que tota persona que tingui un contracte amb qualsevol empresa, és una persona contractada.

Finalment, les restriccions d'integritat definides en aquest exemple estableixen que en cap estat de la base de dades, es pot donar el cas de tenir un empleat que no tingui l'edat mínima legal per treballar; que una persona no pot cobrar un sou en una empresa si no hi treballa; i que una persona que té assignat un número de la Seguretat Social no pot ser que no estigui contractat.

Observi's a més a més, que els atributs que componen la clau de tots els predicats estan subratllats i que no hi ha definida explícitament cap restricció d'integritat de clau. Encara que no hi aparegui explícitament, suposem que tenim definit el predicat d'inconsistència *Ic*. □